Detection and Elimination of Duplicates from Multidatabase Searches*

BY JUNE E. SLACH, *Library Systems Specialist*

Corporate Technical Library
The Upjohn Company
301 Henrietta Street
Kalamazoo, Michigan 49001

ABSTRACT

A major problem in the review and synthesis of citations resulting from multidatabase searching is overlap in coverage among the databases. The ability to identify and eliminate duplicate references from a multidatabase search provides an important service and significant time savings for the end user. The Upjohn Company's Corporate Technical Library has developed software to detect and eliminate duplicates from literature searches captured in electronic format. Methods for the identification of duplicates and the merging and sorting of unique citations are discussed, together with the library's procedures for electronic data capture.

LIBRARIANS and information specialists have for some time been aware of the benefits of electronic data capture and downloading from commercial databases. The availability of literature-search results in electronic form instead of a static paper copy allows us to tackle one of the major problems in multidatabase searching: duplicate citations. This problem is inherent in any cross-database search because of overlap in journal coverage among the various database producers. Literature-search requesters are forced to wade through pages of printouts to condense the many citations into a usable bibliography.

Several solutions have been proposed to the problem of duplicate retrieval. Martyn recommends using a microcomputer and software to create a unique identifier for each citation, sort the references in identifier order, and print the sorted list so that duplicates appear sequentially [1]. Riley, Bell, and Finucane report their use of an intelligent terminal to eliminate duplicates from searches performed on the SDC search system [2].

An ideal answer would be to identify and eliminate duplicates on the vendor's computer, before the printout is produced at the local work station. This solution is proposed by Onorato and Bianchi for legal, economic, and technical reasons [3]. Kegeleers suggest that vendors should offer to suppress duplicates so that searchers are not paying many times for the same information [4]. However, until vendors develop such software, the solution lies at the local searcher's end. If the searcher is using a microcomputer or intelligent terminal for online searching, there may be some data processing capability available with the hardware. It is likely, though, to be fairly unsophisticated and limited in scope. It may not be possible to recognize the many different citation formats retrieved in a typical cross-database search. For this reason, at the Upjohn Company's Corporate Technical Library, literature-search results are transferred to a mainframe computer where more complex manipulations may be performed.

BACKGROUND

For the literature search function, the Corporate Technical Library uses a DEC (Digital Equipment Corporation) WS248 word processing system that has been upgraded to support a hard disk storage unit. The system includes communications hardware and autodial modems for communicating with outside and internal databases. Each of the information specialists searches online using a CRT connected to the central DEC processor. Search results may be recorded on the system's hard disk and recalled to the terminal screen, where editing operations can be performed using the DEC system's word processing capability.

The information specialists at the Corporate Technical Library consult an average of 1.8 databases and 3.6 files for each completed search. Since this is not a high rate of multiplicity, the duplicate-checking process need not be performed on a routine basis. However, every month there are information requests that require access to many

*Based on a paper presented May 29, 1984, at the Eighty-fourth Annual Meeting of the Medical Library Association, Denver, Colorado.

online files for a thorough search of the literature. The library also receives requests for special-purpose bibliographies and projects that involve multidatabase searching for comprehensive summaries of the literature. For example, the library produces bibliographies on particular Upjohn products and generates a yearly bibliography of publications by Upjohn employees. If the duplicate citations could be identified and weeded out before the results are mailed to the requester, there would be significant time savings for the end user. Such a sophisticated computing task cannot be done on the DEC system because of its lack of data processing capabilities. Therefore, a procedure was developed for transmitting the electronic format citations to the Research Division's IBM 3083 mainframe over telephone lines. Once on the mainframe, the captured citations are manipulated by a set of programs written by the library's systems specialist.

Identification of Match Key

Before the captured citations are transmitted to the mainframe computer, the information specialists use edit keys to delete the search commands and other extraneous information. The name of the database from which the citations were retrieved is retained as the first line of information. Citations from each database searched follow in sequence, preceded by the database name. The searcher then dials up the mainframe computer and initiates a program that collects the citations from the DEC system and stores them in a file on the mainframe. The data are then ready to be processed by the duplicate-checking program.

A major problem in writing a duplicate-checking program is determining what constitutes a duplicate. Many data elements have been proposed in various combinations for this purpose. Giles et al. discuss the generation of a "dupcheck" key from citation elements and conclude that year, pages, author, and journal title are the most successful elements in identifying duplicates [5]. Hawkins uses a combination of the CODEN, year, and pagination fields to create an index to citations that identifies a large percentage of duplicates [6]. Hickey reports on the use of a variable-length key to detect duplicate monographic records [7]. The Upjohn Corporate Technical Library has been successful using a key made up of a two-digit year, the first four characters of the first author's name, and the beginning page number. We have found that this key identifies duplicates incorrectly only about 1% of the time. The computer program must locate those three data elements in each citation. This is a complex programming problem because of the wide variety of citation formats.

Search services which tag their data elements with labels (AU, TI, YR) simplify the process of locating a key. The author and year may be readily identified by searching for the AU and YR labels and then extracting the needed information from the same line. The beginning page number must be extracted from the SO (source) line, but each database presents the page numbers in a different location. The duplicate-checking program contains a separate code for each database that has a unique citation format. Databases in the Dialog search system present a more difficult problem because their citations are not tagged with data element labels when they are downloaded or printed. For those citations, the program must locate the key elements by counting indented lines and determining where the author, year, and page numbers are routinely located within the citation. The duplicate-checking program, then, tries to create a match key for each citation, consisting of year, author, and first page number. If any of the three components of the key cannot be located within the citation, the program inserts a blank. If the citations come from a database not known to the program, a blank key is assigned, and a message is printed out so that the programmer may add that database to the program. When all of the citations have been processed, they are sorted in match key order (year, author, page number).

Identification of Duplicates

The computer program identifies the duplicates by looking at two citations at a time and comparing their match keys. If the keys are equivalent, the citations are considered to be duplicates. One citation is kept for the final search result, and the other is stored in a duplicates file. If the searcher wishes, the two citations may be printed out so that they can be double-checked before the bibliography is printed. If a citation is found to have been incorrectly identified as a duplicate, it may be returned to the search results before the final bibliography is produced.

When a duplicate citation has been identified, the program must decide which citation to retain. One method would be to determine which citation has an abstract and retain that one. Another would be to select the longest citation, on the assumption that it would contain the most information. Currently, the searchers at Upjohn assign a priority to each database indicating preference in selection. Citations from a database with a high priority are selected ahead of low-priority citations. The priorities assigned to databases can be changed to fit the requirements of a particular search or project.

SEARCH RESULTS

The result of the duplicate-checking program is a bibliography of unique citations sorted by author or year. Each citation is labeled with the name of the database from which it came. The bibliography may be printed out or transmitted to the search requester by electronic mail. The total processing time for duplicate checking and production of the bibliography is less than five minutes. The greatest amount of time is spent in editing the original search results to eliminate search commands and any other material not handled by the duplicate-checking program. Transmitting the citations to the mainframe computer is also time-consuming. At 1,200 baud, it can take a significant amount of time to transfer several hundred citations. The library systems staff is exploring the possibility of transmittal at 9,600 baud.

The first use of the duplicate-checking program was to create a bibliography on platelet-activating factor (PAF). Eleven files were searched, representing five separate databases (Table 1). A total of 966 citations were retrieved. After the duplicate-checking program was run against the citations, the number of unique citations was reduced to 467, a reduction of 52%. The value of such a program in a project of this size is obvious. To identify all the duplicates manually and eliminate them one by one would have taken many hours of the information specialist's time. The resulting bibliography is clearly superior to the raw printout of 966 citations that could not be sorted or even merged.

TABLE 1
BIBLIOGRAPHY ON PLATELET-ACTIVATING FACTOR

Online File	Citations Retrieved	Unique Citations
ISI/Biomed	242	124
BRS/CHEM, CHEB	183	176
BRS/MESH, MS78 MS74	180	49
BRS/BIOL, BIOB	219	100
Excerpta Medica File 72, 73, 172	142	18
Totals	966	467

CONCLUSION

In thirty-one searches conducted to date, the amount of duplication in any one search has ranged from none to 64%. The percentage varies according to the topic being searched and the databases used. The library systems specialist is presently accumulating statistics for each use of the duplicate-checking program. The statistics include a record of all databases in which a particular citation was located, which databases produced citations found nowhere else, and which were the most productive. A brief description of the search topic is included in the record. Such overlap statistics should help identify the types of searches that would be good candidates for duplicate checking.

The ability to merge and sort disparate citation formats and to identify and eliminate duplicates from the search result is an important tool in the arsenal of the information specialist. Not only does it save time and effort for the searcher and end-user, but it produces an attractive search: one that is a finished product instead of a series of raw printouts. The benefits are already being realized by Upjohn researchers and library users.

REFERENCES

1. Martyn J. Unification of the results of online searches of several databases. Aslib Proc 1982 Aug;34:358–63.
2. Riley C, Bell M, Finucane T. Elimination of duplicate citations from cross database searching using an "intelligent" terminal to produce report-style searches. Online 1981 Oct;5:36–41.
3. Onorato ES, Bianchi G. Automatic identification of duplicates after multidatabase online searching. Online Rev 1981;5:445–51.
4. Kegeleers R. Suggests duplicates be deleted before printout. Online 1983 Jan;7:4.
5. Giles CA, Brooks AA, Doszkocs T, Hummel DJ. An experiment in computer-assisted duplicate checking. Proc Am Soc Inf Sci Ann Meet 1976;13:108.
6. Hawkins D. Machine-readable output from online searches. J Am Soc Inf Sci 1981 July;32:253–6.
7. Hickey TB, Rypka DJ. Automatic detection of duplicate monographic records. J Libr Auto 1979 June;12:125–42.

Received September 1984; accepted December 1984.